

# **OOHDM**

## **Object-Oriented Hypermedia Design Method**

*Eine systematische Methode  
zum Entwurf von Web-Systemen*

Nora Koch

## **Überblick**

- **Einleitung**
- **OOHDM Vorgehensmodell**
- **OOHDM Schritte + Notation**
- **Anwendungsbeispiele: Teach&Learn@Internet  
Projektinformationssystem**
- **Zusammenfassung**

## Was sind Hypermediasysteme?

- **Hypermedia = Hypertext + Multimedia**
- **Hypertext ist ein nicht-lineares Dokument bestehend aus Knoten und Verweisen.**
- **Multimedia: Text, Bild, Ton und Film**
- **Web-Systeme sind Hypermediasysteme, die über das World Wide Web zugänglich sind.**

## Was ist OOHDM ?

**Eine Methode für die Entwurfsphase  
von Hypermediasystemen → Web-Systemen**

**Eine objektorientierte Methode, die das Design  
von Inhalt, Struktur und Präsentation trennt.**

## Methoden zum Entwurf von Hypermediasystemen

<p style="text-align: center;"><b>RMM</b></p> <p style="text-align: center;"><i>Relationship Management Model</i></p> <ul style="list-style-type: none"> <li>• eine Methode für strukturiertes Hypermediadesign</li> <li>• basiert auf dem E-R Modell</li> <li>• T.Isakowitz, E. Stohr, P. Balasubramanian, 1995</li> </ul>	<p style="text-align: center;"><b>HDM</b></p> <p style="text-align: center;"><i>Hypermedia Design Method</i></p> <ul style="list-style-type: none"> <li>• Grob- und Feinentwurf</li> <li>• Trennung von Inhalt, Struktur, Präsentation, Dynamik und Interaktion.</li> <li>• F. Garzotto, L. Mainetti, P. Paolini, 1993</li> </ul>
<p style="text-align: center;"><b>EORM</b></p> <p style="text-align: center;"><i>Enhanced Object-Relationship Model</i></p> <ul style="list-style-type: none"> <li>• interaktionsorientiertes Modell</li> <li>• basiert auf OMT</li> <li>• D. Lange, 1994</li> </ul>	<p style="text-align: center;"><b>OOHDM</b></p> <p style="text-align: center;"><i>Object-Oriented Hypermedia Design</i></p> <ul style="list-style-type: none"> <li>• objektorientierter Ansatz</li> <li>• basiert auf OMT</li> <li>• D. Schwabe, G. Rossi, 1995</li> </ul>

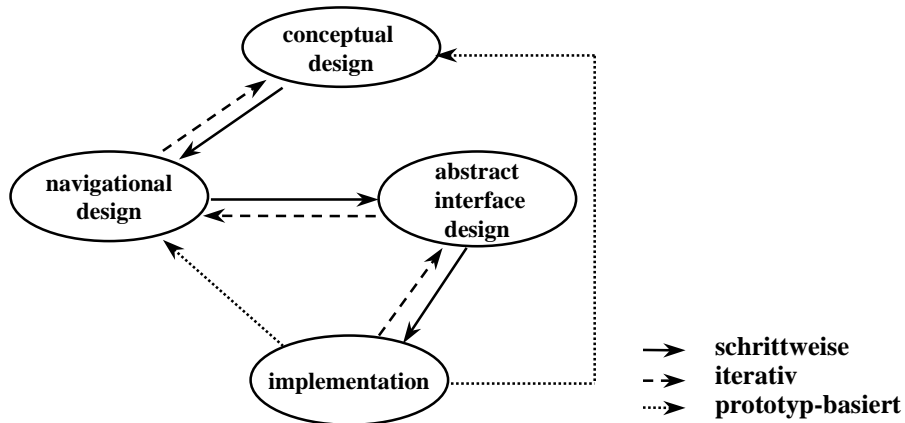
## OOHDM Vorgehensmodell

### Entwurf

- des Anwendungsgebiets: *Conceptual Design*
- der Struktur und Verknüpfungen: *Navigational Design*
- der Präsentation: *Abstract Interface Design*

**Implementierung:** *Implementation*

## OOHDM-Entwurfsprozeß



## OOHDM Schritte

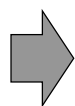
Schritte	Ziele	Mechanismen	Darstellung	Produkte
<b>Conceptual Design</b>	Modellierung der Semantik des Anwendungsgebiet	Objektorientierten Prinzipien		
<b>Navigational Design</b>	Modellierung der Struktur unter Berücksichtigung von kognitiven, benutzerspezifischen Aspekten	Abbildung von Conceptual- auf Navigations- Objekte		
<b>Abstract Interface Design</b>	Modellierung der ausgewählten Metaphern und wahrnehmbare Objekte	Abbildung von Navigations- auf wahrnehmbare Objekten		
<b>Implementa-tion</b>	Vollständigkeit und Performance	Von den Werkzeugen zur Verfügung gestellt		

## Beispiel: Teach&Learn@Internet

- Schulen + Partnerschulen
- gemeinsame Projekte
- Projektgruppe: Lehrer + Schüler
- Rollen: Projektteilnehmer, Projektleiter, zentraler und lokaler Administrator, Leser
- multimediale Dokumente
- Protokolle dokumentieren die Entwicklung des Projektes
- Kommunikation: E-Mail + Chatforen
- Steckbriefe
- Mehrsprachigkeit
- Metapher: Öffentlicher und nicht öffentlicher Bereich  
Lehrerzimmer, Klassenzimmer und Pausenhof

## Conceptual Design

- Erstellung eines Modells für das Anwendungsgebiet
- unabhängig von Benutzeraufgaben und -zielen
- unabhängig von der Darstellung

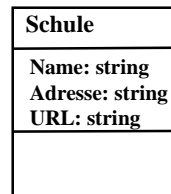


*Objektorientierte Methode OMT  
erweitert  
mit Attributsperspektiven und Subsysteme*

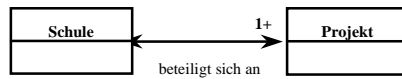
**OMT: Object Modeling Technique** (Rumbaugh)  
Objektmodell, dynamisches Modell und funktionales  
Modell

## OMT Notation

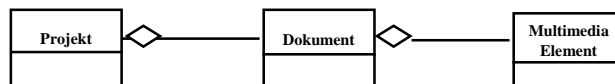
**Klasse:** Name + Attribute + Methoden



**Assoziation:**

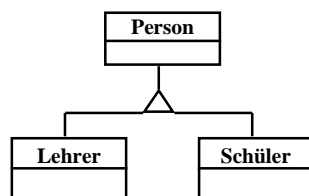


**Aggregation:**

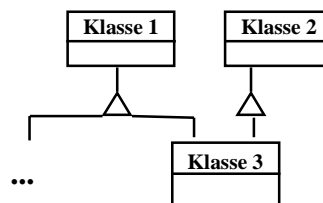


## OMT Notation

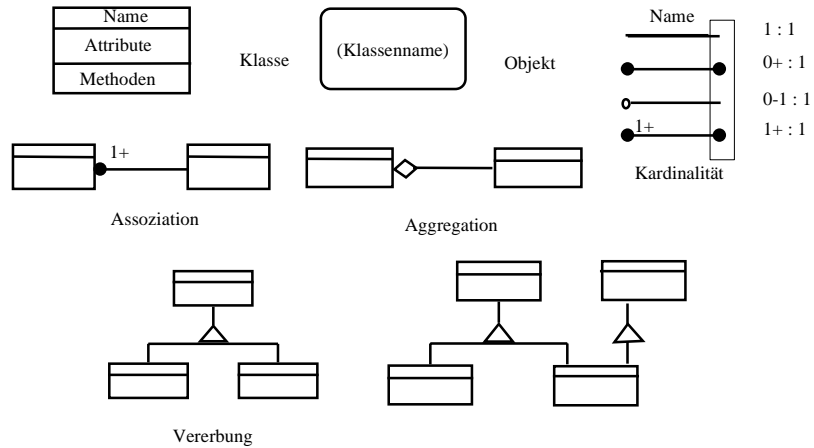
**Vererbung**



**Mehrfachvererbung**



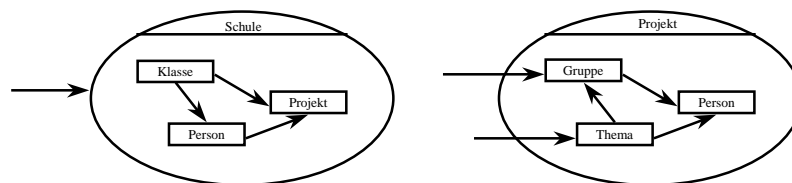
## Zusammenfassung der OMT Notation



## OMT-Erweiterungen

### *Subsysteme*

mit einem oder mehreren Einstiegspunkten



### *Attributsperspektiven*

Attribute sind nicht auf einen Typ beschränkt, z.B.

Beschreibung: [bild, text+,video] + Default-Typ

## Conceptual Design: Aktivitäten

- Definition von Subsysteme, Klassen, Attribute, Methoden
- Zuweisungen von Typen zu Attributen
- Feststellung von Assoziations- und Aggregationsbeziehungen
- Darstellung der Vererbungshierarchie

• Objektmodell  *CONCEPTUAL SCHEMA*

## Schablonen

- Subsystem
- Klasse
- Beziehung
- Objekt

	subclass name	
	classes included	
	relations	classes
	related with	
	entry points	
	comments	
	back trace	forward trace



## Schablone

	class name <b>Schule</b>	
	attributes <b>name:str</b> <b>anschrift:str</b> <b>url:str</b>	
	methods	
	relations related with	classes
	<b>partnerschulen</b> <b>beteiligung</b> <b>zugehörigkeit</b>	<b>Schule()</b> <b>Projekt()</b> <b>Gruppe()</b>
	part of	
	inherits from	
	comments	
	back trace	forward trace

## Navigational Design

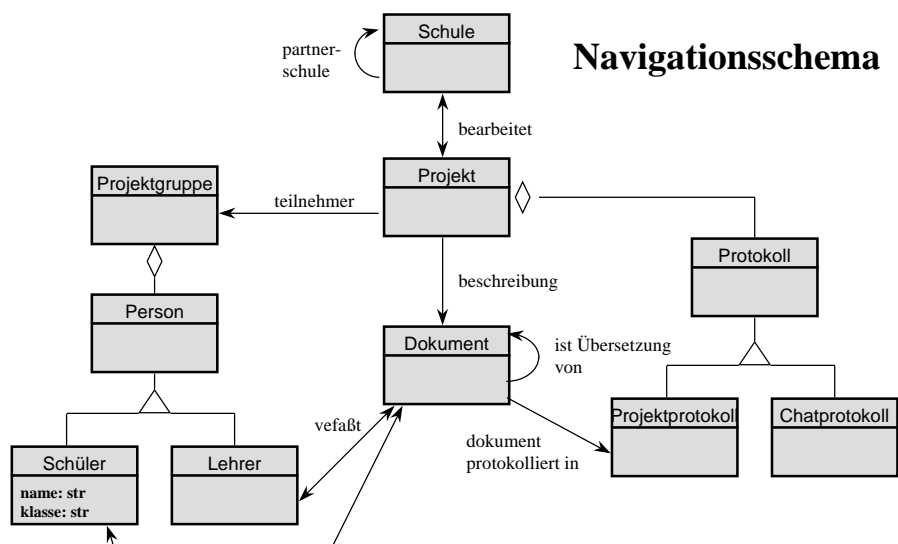


**Modellierung der Navigationsstruktur unter Berücksichtigung von kognitiven, benutzerspezifischen Aspekten.**

- **Definition von**
  - **Knoten und Links**
  - **Navigationskontexten und Klassen zur Kontextdekoration**
  - **Zugriffsstrukturen**
- **Graphische Darstellung im**
  - **Navigationsschema** (navigational schema)
  - **Kontextschema** (contextual schema)
  - **Navigationsdiagramm** (navigational diagram or chart)

## Navigationsschema

- ist ein Grobentwurf der Navigationsstruktur einer hypermedialen Anwendung
- besteht aus Navigationsknoten und Links.
  - Navigationsknoten werden von den Klassen des Conceptual Modells definiert.
  - Links beschreiben die nicht-linearen Verknüpfungen und basieren auf Beziehungen des Conceptual Modells.
- wird mit einer OMT-ähnlichen Notation graphisch dargestellt.



## Schablonen für Navigationsklassen

Class Node:				of			
attributes:							
methods:							
		class	by links	anchor			
related with:							
appear in contexts:							
part of:							
inherits from:							
comments:							
back trace:				forward trace:			

Class Link:		of	
source:			
target:			
cardinality:			
attributes:			
target link:			
transformation:		refined in:	
comments:			
back trace:		forward trace:	

## Kontexte

Für ein Anwendungsgebiet können mehrere Navigationsmodelle definiert werden, es sind unterschiedliche Sichten auf das Anwendungsgebiet (Konzept aus "database views")



In diesem Schritt wird eine verfeinerte Navigationsstruktur auf der Basis von Navigationskontexten und der Zugriffsstrukturen konstruiert.

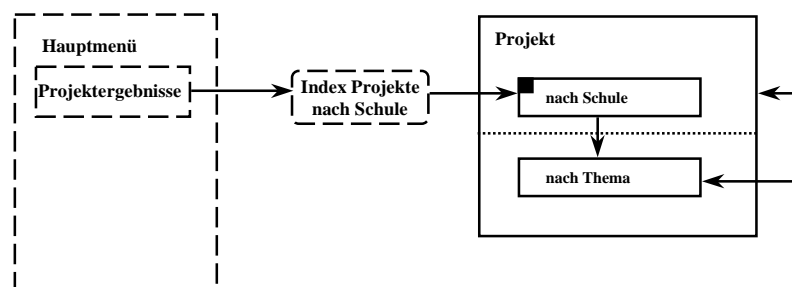
- *Navigationskontexte* bestehen aus Knoten, Links und eventuell aus anderen Kontexten und beinhalten Pfade zwischen den Objekten.
- *Zugriffsstrukturen* wie ein Index oder eine Führung haben eine definierte Semantik und legen den Einstiegspunkt des Kontextes fest.

## Wie werden Navigationskontexte gefunden?

- **von Klassen hergeleitet:** alle Objekte einer Klasse oder alle, die eine Bedingung erfüllen, z.B. Projekte, die sich mit dem Thema Umwelt beschäftigen.
- **von einer Beziehung hergeleitet:** alle Objekte, die in einer Beziehung zu einer Klasse stehen, z.B. alle Protokolle eines Projektes.
- **durch Aufzählung:** eine explizit definierte Menge, z.B. eine Führung.
- **Antwort auf Anfrage:** Elemente werden auf Basis einer Benutzeranfrage definiert.
- **dynamisch generiert:** die Menge ändert sich während der Navigation, z.B. Einkaufskorb.

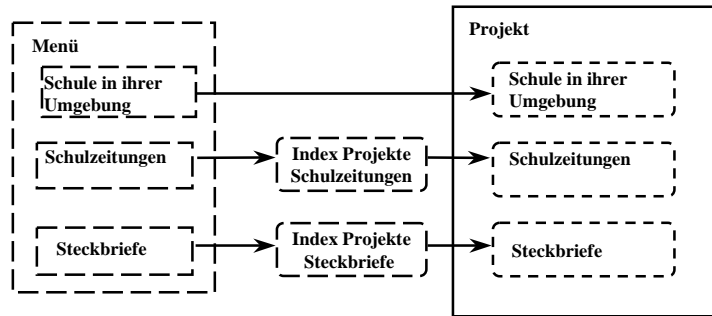
## Kontextschema (1)

Teach&Learn@Internet-Anwendung



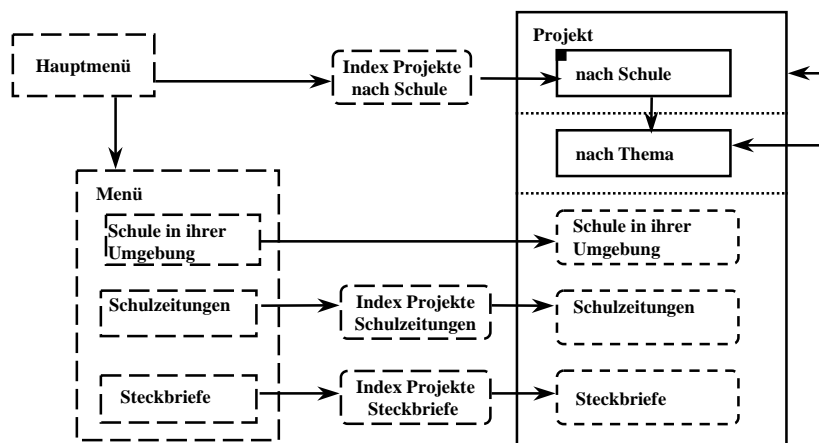
## Kontextschema (2)

Teach&Learn@Internet-Anwendung

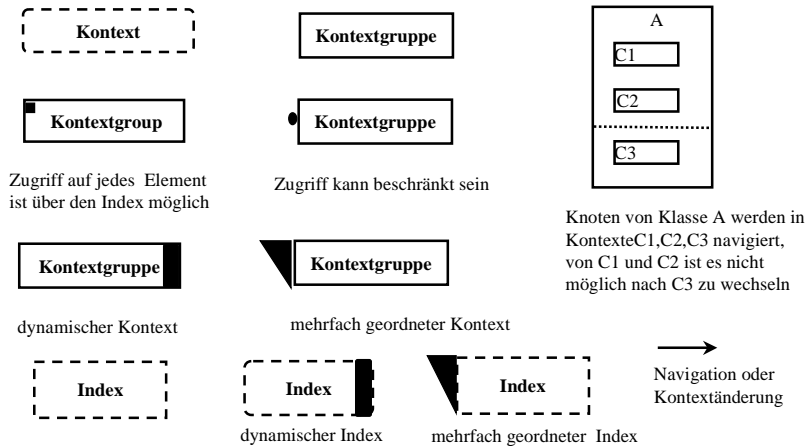


## Kontextschema (3)

Teach&Learn@Internet-Anwendung

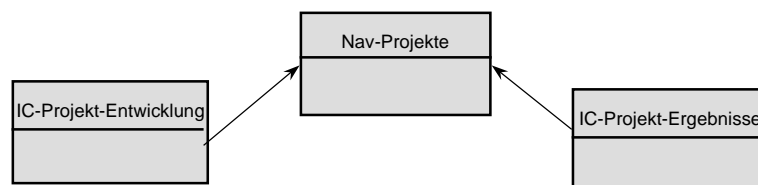


## Notation für Kontextschemata



## InContext Klassen

- **Knoten können für jeden Kontext zusätzliche Attribute und/oder anderes Verhalten bzw. andere Anker besitzen**
- **InKontext Klassen “dekorieren” Kontexte**



## Schablonen für Kontextklasse und Zugriffstruktur

Context Class name:	type:
attributes:	
include:	classes InContext
context changes:	
entry points:	
initial entry point:	
path:	
link type in context:	
comments:	
back trace:	forward trace:

Access structure name:	
inherits from:	
selectors:	
structure:	
target:	
attributes:	
behaviour:	
comments:	
back trace:	forward trace:

## Navigation Chart

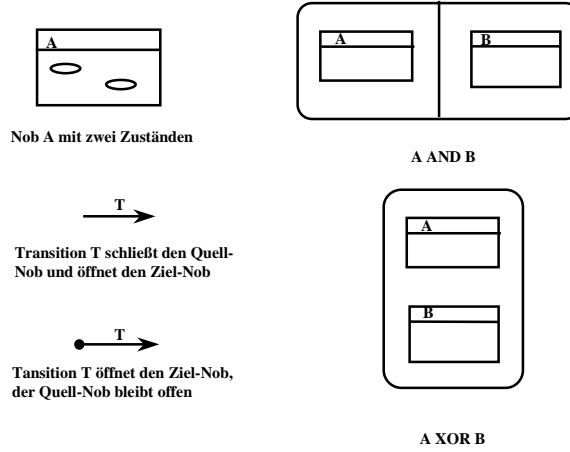


**In diesem Schritt werden die dynamischen Aspekte der Navigation in einem StateChart-ähnlichen Diagramm modelliert.**

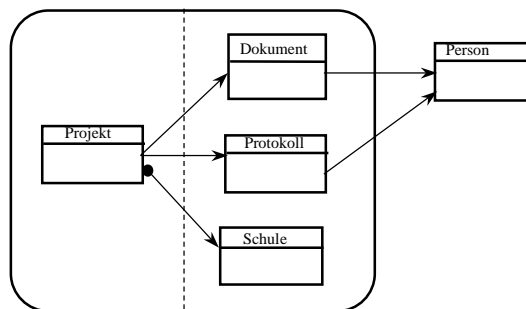
**Hierfür wird ein Navigationsdiagramm (Navigation Chart) erstellt, das mit Navigationsobjekten, Zuständen und Transitionen konstruiert wird.**

- *Navigational objects (Nobs)* sind Knoten und Zugriffsstrukturen.
- Ein *Cluster* wird in OOHDM definiert, um Objekte zu identifizieren, die im selben Navigationsraum koexistieren können.
- **Nobs und Cluster können verschachtelt sein.**

## Notation für Navigationsdiagramme



## Navigationsdiagramm





## Abstract Interface Design



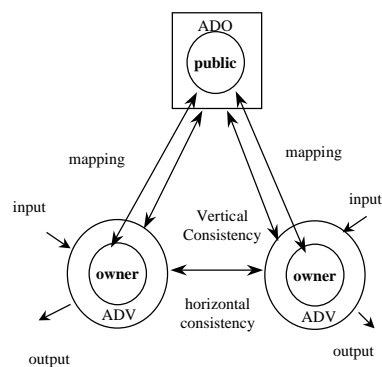
**Modellierung der statischen und dynamischen Aspekte der Benutzerschnittstelle**

Hierfür werden folgende Elemente spezifiziert:

- das Layout der Navigationsobjekte (unterschiedliches Layout je nach Kontext)
- Schnittstellenelementen mit Interaktionsfunktionalität wie Knöpfe und Menüs
- Beziehung zwischen Navigations- und Schnittstellenobjekten mit Hinweis auf den Einfluß von Benutzerinteraktionen
- Synchronisation von zeitabhängigen Schnittstellenobjekten wie Ton und Film.

## Abstract Data Views

OOHDM verwendet das Abstract Data View (ADV) Konzept um das Abstract Interface Model zu definieren.



- ADVs sind Schnittstellenklassen.
- ADVs besitzen Struktur und Verhalten
- ADVs sind abstrakt weil sie nur den Zustand und die Schnittstelle darstellen und nicht die Implementierung.
- ADVs können als aktive Objekte, ADOs als passive Elemente betrachtet werden
- ein ADO "besitzt" mehrere ADVs

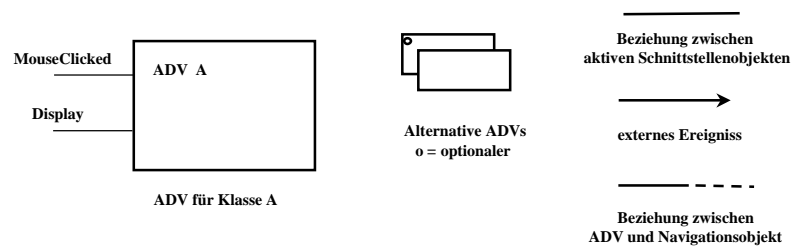
## ADVs in OOHDM

ADVs werden im OOHDM Hypermedia-entwurf verwendet, um die vom Benutzer wahrnehmbaren Eigenschaften der Navigationsobjekte (ADOs) darzustellen.

Wie wird im Abstract Interface Design vorgegangen?

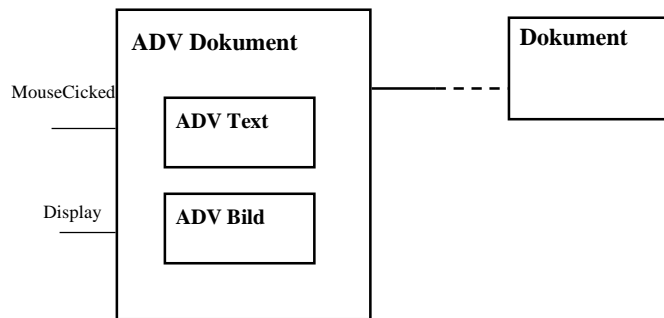
- Definition einer Schnittstellenmetapher
- Definition von mindestens einem ADV für jeden Navigationsknoten, Index und für jeden Kontext
- Spezifikation des ADV-Konfigurationsdiagramms
- Spezifikation von ADV Charts

## Notation für ADV Konfigurationsdiagramme

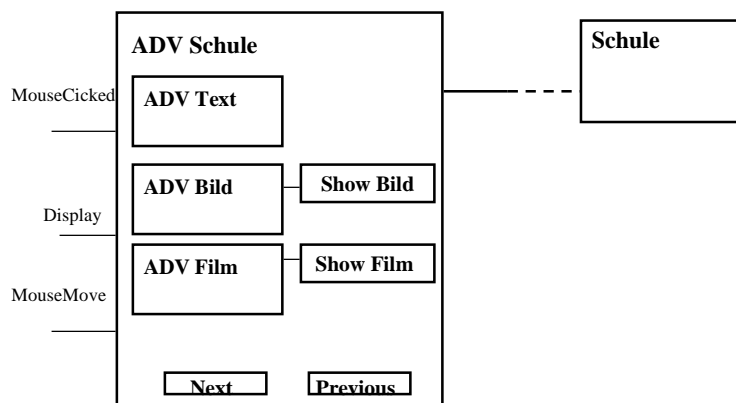


**Komposition- und Abstraktionmechanismen werden im ADV-Entwurfsansatz unterstützt**

## Konfigurationsdiagramme (1)

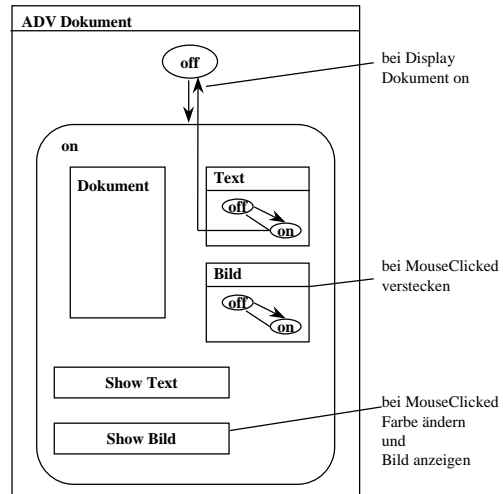


## Konfigurationsdiagramme (2)



# ADV Charts

- ADV Charts sind eine Erweiterung von StateCharts.
- Ziel ist externe Ereignisse zu modellieren
- Notation wie bei Navigationsdiagramm



# Implementierung

Anwendung	Analyse	Entwurf	Implementierung
Portinari-Project		OOHDM	ToolBook Ped
Teach&Learn@ Internet	Use Cases	OOHDM	Hyperwave/ Place
Projekt-informations-system	Use Cases	OOHDM	WebObjects

## Projektinformationssystem

- **Information, um laufende Vertriebsprojekte zu verwalten**
- **Eingabe von Projektvorschlägen**
- **wenn bestimmte Bedingungen erfüllt sind, entsteht aus einem Projektvorschlag ein Projekt**
- **für jedes Projekt werden Meilensteine definiert, Aktivitäten geplant und Notizen eingegeben**
- **Projekte werden von Händler, Partner und Mitarbeiter durchgeführt**
- **Personen können unterschiedliche Rollen bei einem Projekt spielen**
- **Information aus diesem System wird im Forecast- und Trackingsystem verwendet**
- **Erinnerungen an Termine sowie neue Informationen, die von anderen Personen eingegeben worden sind, werden automatisch angezeigt**

## Erfahrungen beim Entwurf mit OOHDM

### Vorteile

- **klare Trennung von Inhalt, Struktur und Präsentation**
- **kontext-basierte Lösungen**
- **systematischer Ansatz für den Entwurf von Hypermediasysteme**

### Schwierigkeiten

- **zu wenig Literatur zu OOHDM in englisch**
- **laufende Verbesserungen und Änderungen in der Notation**
- **keine Unterstützung durch Werkzeuge**

## OOHDM

Schritte	Ziele	Mechanismen	Darstellung	Produkte
<b>Conceptual Design</b>	Modellierung der Semantik des Anwendungsgebiet	Objektorientierten Prinzipien	Conceptual Schema	Klassen, Beziehungen, Sub-Systeme, ...
<b>Navigational Design</b>	Modellierung der Struktur unter Berücksichtigung von kognitiven, benutzerspezifischen Aspekten	Abbildung von Conceptual- auf Navigations- Objekte	Navigational Schema, Contextual Schema, Navigational Charts	Knoten, Links, Zugriffsstrukturen, Navigationskontexte
<b>Abstract Interface Design</b>	Modellierung der ausgewählten Metaphern und wahrnehmbaren Objekte	Abbildung von Navigations- auf wahrnehmbare Objekte	Abstract Data Views Configuration Diagrams, ADV Charts	Abstrakte Präsentations- und Interaktionsobjekte
<b>Implementation</b>	Vollständigkeit und Performance	Von der Umgebung zur Verfügung gestellt	Von der Umgebung zur Verfügung gestellt	Lauffähige Anwendung